Readings:
K&F: 15.1, 15.2, 15.3, 15.4, 15.5

Structure Learning: the good, the bad, the ugly

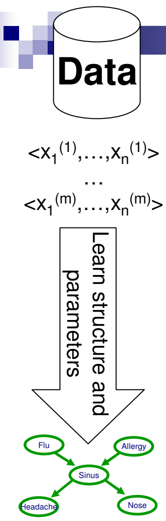
Graphical Models – 10708
Carlos Guestrin
Carnegie Mellon University

September 29th, 2006

Where are we with learning BNs?

- Given structure, estimate parameters
  - Maximum likelihood estimation
  - Bayesian learning
- What about learning structure?

Learning the structure of a BN



- **Constraint-based approach**
  - BN encodes conditional independencies
  - Test conditional independencies in data
  - Find an I-map
- **Score-based approach**
  - Finding a structure and parameters is a density estimation task
  - Evaluate model as we evaluated parameters
    - Maximum likelihood
    - Bayesian
    - etc.

Remember: Obtaining a P-map?

- Given the independence assertions that are true for *P*
  - Obtain skeleton
  - Obtain immoralities
- From skeleton and immoralities, obtain every (and any) BN structure from the equivalence class

- **Constraint-based approach**:
  - Use Learn PDAG algorithm
  - Key question: **Independence test**

Independence tests

- Statistically difficult task!
- Intuitive approach: **Mutual information**

$$I(X_i, X_j) = \sum_{x_i, x_j} P(x_i, x_j) \log \frac{P(x_i, x_j)}{P(x_i)P(x_j)}$$

- Mutual information and independence:
  - X_i and X_j independent if and only if $I(X_i,X_j)=0$
- Conditional mutual information:

Independence tests and the constraint based approach

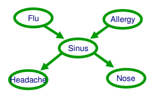
- Using the data *D*
  - Empirical distribution: $\widehat{P}(x_i, x_j) = \frac{\text{Count}(x_i, x_j)}{m}$
  - Mutual information: $\widehat{I}(X_i, X_j) = \sum_{x_i, x_j} \widehat{P}(x_i, x_j) \log \frac{\widehat{P}(x_i, x_j)}{\widehat{P}(x_i)\widehat{P}(x_j)}$
  - Similarly for conditional MI
- Use learning PDAG algorithm:
  - When algorithm asks: $(X \perp Y \mid \mathbf{U})$?
- Many other types of independence tests
  - See reading…

Score-based approach

Data

<$x_1^{(1)},\ldots,x_n^{(1)}$>
…
<$x_1^{(m)},\ldots,x_n^{(m)}$>

Possible structures

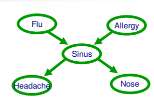


Learn parameters

Score structure

Information-theoretic interpretation of maximum likelihood

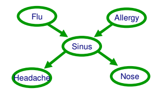
- Given structure, log likelihood of data:



$$\log P(\mathcal{D} \mid \theta_{\mathcal{G}}, \mathcal{G}) = \sum_{j=1}^{m} \sum_{i=1}^{n} \log P\left(X_i = x_i^{(j)} \mid \mathbf{Pa}_{X_i} = \mathbf{x}^{(j)}\left[\mathbf{Pa}_{X_i}\right]\right)$$

Information-theoretic interpretation of maximum likelihood 2

- Given structure, log likelihood of data:



$$\log \hat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = m \sum_i \sum_{x_i, \mathbf{Pa}_{x_i,\mathcal{G}}} \hat{P}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) \log \hat{P}(x_i \mid \mathbf{Pa}_{x_i,\mathcal{G}})$$

Decomposable score

- Log data likelihood

$$\log \hat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = m \sum_i \hat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - M \sum_i \hat{H}(X_i)$$

- Decomposable score:
  - Decomposes over families in BN (node and its parents)
  - Will lead to significant computational efficiency!!!
  - Score($G : D$) = $\sum_i$ FamScore($X_i|\mathbf{Pa}_{X_i} : D$)

How many trees are there?

Nonetheless – Efficient optimal algorithm finds best tree

Scoring a tree 1: I-equivalent trees

$$\log \widehat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = M \sum_i \widehat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - M \sum_i \widehat{H}(X_i)$$

Scoring a tree 2: similar trees

$$\log \widehat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = M \sum_i \widehat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - M \sum_i \widehat{H}(X_i)$$

Chow-Liu tree learning algorithm 1

- For each pair of variables X_i,X_j
  - Compute empirical distribution:

$$\widehat{P}(x_i, x_j) = \frac{\text{Count}(x_i, x_j)}{m}$$

  - Compute mutual information:

$$\widehat{I}(X_i, X_j) = \sum_{x_i, x_j} \widehat{P}(x_i, x_j) \log \frac{\widehat{P}(x_i, x_j)}{\widehat{P}(x_i)\widehat{P}(x_j)}$$

- Define a graph
  - Nodes $X_1,\ldots,X_n$
  - Edge (i,j) gets weight $\widehat{I}(X_i, X_j)$

Chow-Liu tree learning algorithm 2

$$\log \hat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = M \sum_i \hat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - M \sum_i \hat{H}(X_i)$$

- Optimal tree BN
  - Compute maximum weight spanning tree
  - Directions in BN: pick any node as root, breadth-first-search defines directions

Can we extend Chow-Liu 1

- Tree augmented naïve Bayes (TAN) [Friedman et al. '97]
  - Naïve Bayes model overcounts, because correlation between features not considered
  - Same as Chow-Liu, but score edges with:

$$\hat{I}(X_i, X_j \mid C) = \sum_{c, x_i, x_j} \hat{P}(c, x_i, x_j) \log \frac{\hat{P}(x_i, x_j \mid c)}{\hat{P}(x_i \mid c)\hat{P}(x_j \mid c)}$$

Can we extend Chow-Liu 2

- (Approximately learning) models with tree-width up to *k*
  - [Narasimhan & Bilmes '04]
  - But, $O(n^{k+1})$…
    - and more subtleties

What you need to know about learning BN structures so far

- Decomposable scores
  - Maximum likelihood
  - Information theoretic interpretation
- Best tree (Chow-Liu)
- Best TAN
- Nearly best k-treewidth (in $O(N^{k+1})$)

Announcements

- Homework 2 out
  - Due Oct. 11th
- Project description out next week

Maximum likelihood score overfits!

$$\log \widehat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = M \sum_i \widehat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - M \sum_i \widehat{H}(X_i)$$

- Information never hurts:
- Adding a parent always increases score!!!

Bayesian score

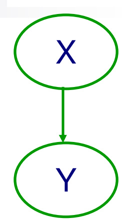
- Prior distributions:
  - Over structures
  - Over parameters of a structure
- Posterior over structures given data:

$$\log P(\mathcal{G} \mid D) \propto \log P(\mathcal{G}) + \log \int_{\theta_{\mathcal{G}}} P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) P(\theta_{\mathcal{G}} \mid \mathcal{G}) d\theta_{\mathcal{G}}$$

Bayesian score and model complexity

$$\log P(D \mid \mathcal{G}) = \log \int_{\theta_{\mathcal{G}}} P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) P(\theta_{\mathcal{G}} \mid \mathcal{G}) d\theta_{\mathcal{G}}$$

True model:



P(Y=t|X=t) = 0.5 + α
P(Y=t|X=f) = 0.5 - α

- Structure 1: X and Y independent
  - Score doesn't depend on alpha
- Structure 2: X $\rightarrow$ Y
  - Data points split between P(Y=t|X=t) and P(Y=t|X=f)
  - For fixed M, only worth it for large α
    - Because posterior over parameter will be more diffuse with less data

Bayesian, a decomposable score

$$\log P(D \mid \mathcal{G}) = \log \int_{\theta_{\mathcal{G}}} P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) P(\theta_{\mathcal{G}} | \mathcal{G}) d\theta_{\mathcal{G}}$$

- As with last lecture, assume:
  - Local and global parameter independence
- Also, prior satisfies **parameter modularity**:
  - If X_i has same parents in *G* and *G'*, then parameters have same prior
- Finally, structure prior P(*G*) satisfies **structure modularity**
  - Product of terms over families
  - E.g., $P(G) \propto c^{|G|}$
- Bayesian score decomposes along families!

BIC approximation of Bayesian score

- Bayesian has difficult integrals
- For Dirichlet prior, can use simple Bayes information criterion (BIC) approximation
  - In the limit, we can forget prior!
  - **Theorem**: for Dirichlet prior, and a BN with Dim(*G*) independent parameters, as $M \to \infty$:

$$\log P(D \mid \mathcal{G}) = \log P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) - \frac{\log M}{2} \text{Dim}(\mathcal{G}) + O(1)$$

BIC approximation, a decomposable score

- BIC: $\text{Score}_{\text{BIC}}(\mathcal{G} : D) = \log P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) - \frac{\log M}{2}\text{Dim}(\mathcal{G})$

- Using information theoretic formulation:

$$\text{Score}_{\text{BIC}}(\mathcal{G} : D) = M \sum_i \hat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - M \sum_i \hat{H}(X_i) - \frac{\log M}{2} \sum_i \text{Dim}(P(X_i \mid \mathbf{Pa}_{x_i,\mathcal{G}}))$$

Consistency of BIC and Bayesian scores

Consistency is limiting behavior, says nothing about finite sample size!!!

- A scoring function is **consistent** if, for true model G^*, as $M \rightarrow \infty$, with probability 1
  - G^* maximizes the score
  - All structures **not I-equivalent** to G^* have strictly lower score
- **Theorem**: BIC score is consistent
- **Corollary**: the Bayesian score is consistent
- What about maximum likelihood score?

Priors for general graphs

- For finite datasets, prior is important!
- Prior over structure satisfying prior modularity

- What about prior over parameters, how do we represent it?
  - *K2 prior*: fix an α, $P(\theta_{X_i|\mathbf{Pa}_{X_i}}) = \text{Dirichlet}(\alpha,\ldots,\alpha)$
  - K2 is "inconsistent"

BDe prior

- Remember that Dirichlet parameters analogous to "fictitious samples"
- Pick a fictitious sample size m'
- For each possible family, define a prior distribution $P(X_i,\mathbf{Pa}_{X_i})$
  - Represent with a BN
  - Usually independent (product of marginals)
- **BDe prior**:

- Has "consistency property":

Score equivalence

- If *G* and *G'* are I-equivalent then they have same score
- **Theorem 1**: Maximum likelihood score and BIC score satisfy score equivalence
- **Theorem 2**:
  - If P(*G*) assigns same prior to I-equivalent structures (e.g., edge counting)
  - and parameter prior is dirichlet
  - then **Bayesian score satisfies score equivalence** *if and only if* prior over parameters represented as a **BDe prior**!!!!!!

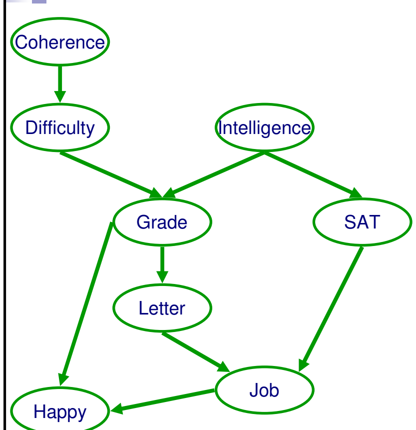
Chow-Liu for Bayesian score

- Edge weight $w_{X_j \to X_i}$ is advantage of adding X_j as parent for X_i
- Now have a directed graph, need directed spanning forest
  - Note that adding an edge can hurt Bayesian score – choose forest not tree
  - But, if score satisfies score equivalence, then $w_{X_j \to X_i} = w_{X_j \to X_i}$!
  - Simple maximum spanning forest algorithm works

Structure learning for general graphs

- In a tree, a node only has one parent
- **Theorem**:
  - The problem of learning a BN structure with at most *d* parents is NP-hard for any (fixed) $d \geq 2$
- Most structure learning approaches use heuristics
  - Exploit score decomposition
  - (Quickly) Describe two heuristics that exploit decomposition in different ways

Understanding score decomposition



Fixed variable order 1

- Pick a variable order $\prec$
  - e.g., $X_1,\ldots,X_n$
- X_i can only pick parents in $\{X_1,\ldots,X_{i-1}\}$
  - Any subset
  - Acyclicity guaranteed!
- Total score = sum score of each node

Fixed variable order 2

- Fix max number of parents to k
- For each *i* in order $\prec$
  - Pick $\mathbf{Pa}_{Xi} \subseteq \{X_1,\ldots,X_{i-1}\}$
    - Exhaustively search through all possible subsets
    - $\mathbf{Pa}_{Xi}$ is maximum $\mathbf{U} \subseteq \{X_1,\ldots,X_{i-1}\}$ FamScore($X_i|\mathbf{U}$: *D*)
- Optimal BN for each order!!!
- Greedy search through space of orders:
  - E.g., try switching pairs of variables in order
  - If neighboring vars in order are switch, only need to recompute score for this pair
    - O(n) speed up per iteration
    - Local moves may be worse

Learn BN structure using local search

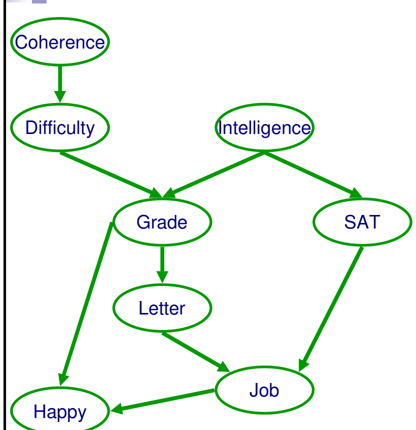
Starting from Chow-Liu tree

Local search,
possible moves:
Only if acyclic!!!
- Add edge
- Delete edge
- Invert edge

Select using favorite score

10-708 – ©Carlos Guestrin 2006 35

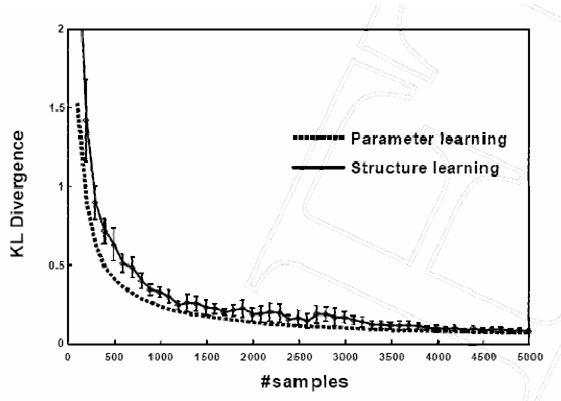
Exploit score decomposition in local search



- Add edge and delete edge:
  - Only rescore one family!
- Reverse edge
  - Rescore only two families

10-708 – ©Carlos Guestrin 2006 36

Some experiments



Alarm network

Order search versus graph search

- Order search advantages
  - For fixed order, optimal BN – more "global" optimization
  - Space of orders much smaller than space of graphs
- Graph search advantages
  - Not restricted to k parents
    - Especially if exploiting CPD structure, such as CSI
  - Cheaper per iteration
  - Finer moves within a graph

Bayesian model averaging

- So far, we have selected a single structure
- But, if you are really Bayesian, must average over structures
  - Similar to averaging over parameters

$$\log P(D \mid \mathcal{G}) = \log \int_{\theta_{\mathcal{G}}} P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) P(\theta_{\mathcal{G}} | \mathcal{G}) d\theta_{\mathcal{G}}$$

- Inference for structure averaging is very hard!!!
  - Clever tricks in reading

What you need to know about learning BN structures

- Decomposable scores
  - Data likelihood
  - Information theoretic interpretation
  - Bayesian
  - BIC approximation
- Priors
  - Structure and parameter assumptions
  - BDe if and only if score equivalence
- Best tree (Chow-Liu)
- Best TAN
- Nearly best k-treewidth (in $O(N^{k+1})$)
- Search techniques
  - Search through orders
  - Search through structures
- Bayesian model averaging